

OTN **OpenMail Technical Note**

B.06.00 Unix ID and OpenMail ID Administration In OpenMail

Summary

At B.06.00, you no longer need to create Unix users before configuring the associated OpenMail user. This OTN looks at the user administration commands and interfaces affected by this change.

This OTN will form part of the OM360 (B.06.00 upgrade) course.

Readership

Readers are assumed to be OpenMail support engineers, who have attended an advanced OpenMail course (AE351 or H1805s) or are familiar with the material covered in these courses.

Revision History

August 1999: First issue

Comments Please!

I would welcome any comments you may have on this document. Please email them to joyce@pwd.hp.com.

Contents

Introduction.....	3
Creating The Pool Of Unix IDs.....	4
Changes To omaddu.....	6
Changes To omshowu	6
Changes To Areas Which Required UNIX Login Names.....	9
Trace And User Tweak Filenames	9
Home Directory For omadmin	9
OpenMail Passwords Linked to Unix Passwords	9
Auditing.....	10
Administration UIs	10
ITO/ITA.....	10
Advmail Terminal Interface	11
Print Server.....	11

Introduction

In previous releases, you had to create a Unix user before creating the corresponding OpenMail user. This meant that each OpenMail user needed to have:

- A Unix logon name, for example, jackl, root.
- A Unix ID, that is, a Unix user number, for example, 500, 30020.
- An OpenMail ID, that is, a number allocated to an OpenMail user.

At B.06.00, the administration of Unix IDs for OpenMail users has been simplified.

- Each OpenMail user still needs to have a Unix ID number, but ordinary users no longer need a Unix logon name.
- For ordinary users, without administrator privileges, you no longer need to create a Unix user before creating the corresponding OpenMail user. Instead, a Unix ID number can be automatically assigned from a pool of numbers, when you create an OpenMail user.
- An OpenMail administrator user still needs to be linked to a proper Unix user, with a Unix logon name, so that they can run administrator commands from a standard shell.

In effect, dummy Unix accounts are created for ordinary, non-admin OpenMail users. These Unix IDs appear in the OpenMail USERLIST entries but not in any Unix password file entries.

It is the responsibility of the OpenMail (and system) administrators to decide which Unix IDs should be allocated for OpenMail users and to set up the pool of Unix ID numbers.

The main reasons for the changes are to simplify the management of non-administrator users and to improve performance. Performance is improved, because OpenMail no longer needs to read a user's password file entry to map OpenMail user IDs to Unix login names for certain operations, such as assigning IDs to user processes and configuring UAL trace files . We will look later at areas which have been changed to remove reliance on the existence of Unix login names.

In this OTN, then, we will look at

- How to create a pool of Unix IDs.
- How to add users with IDs assigned automatically from this pool.
- Areas which have been changed, because they previously used the user's Unix login name.

Creating The Pool Of Unix IDs

If you want to be able to set up OpenMail users using pool IDs, you first have to set up the pool of Unix ID numbers using the command `omadmidp` (**om admin id pool**). The pool file, `/var/opt/openmail/sys/idpool`, does not exist until you use the `omadmidp` command.

```
Usage:  omadmidp -a -s <StartId> [ -n <Range> ]
        omadmidp -d -s <StartId> [ -n <Range> ]
        omadmidp -u [ -f ] [ -v | -q ]
```

This command will allow a range of Unix IDs to be allocated to OpenMail. It will not check the password file, so on systems where processes can be invoked without an entry in the `/etc/passwd` file for the user's Unix ID, no entries in the `/etc/passwd` file need be created.

- a Add a block of Unix ID numbers to the pool, starting at *StartId* and extending for *Range*.
- d Delete Unix ID numbers from the pool starting at *StartId* and extending for *Range*.
- s The first ID number in the range.
- n Number of IDs to be allocated or deleted. If this is not specified, the single ID specified in the `-s` parameter is assumed.
- u List simple statistics; for each block of consecutive IDs in the pool, the statistics shown are the first ID in each block, the range of the block and the number of IDs already used.
- f Fix mode. This option fixes inconsistencies between the ID pool and the USERLIST. If, for some reason, IDs in the pool are marked as used but no corresponding entry is found in the USERLIST, the pool ID will be deallocated.
- q Quick Mode. Don't bother to confirm details with the USERLIST.
- v Verbose. Displays a table showing Unix ID to OpenMail ID mapping.

So to allocate two blocks of 3000 Unix IDs to OpenMail, starting at 100000 and 550000:

```
omadmidp -a -s 100000 -n 3000
omadmidp -a -s 550000 -n 3000
```

To check how many IDs in each block are used:

```
root@kuda[] #omadmidp -u

Start of block: 100000
Range of block:   3000
Used in block:    0
Start of block: 550000
Range of block:   3000
Used in block:    0
```

You can create as many blocks of numbers as you want, as long as they do not overlap. You can also delete a block, provided none of the IDs in the range are in use.

A hidden option, `omadmidp -j`, is used by the admin GUI scripts to find the total number of free entries in the pool.

For a list of which pool IDs are in use and their corresponding OpenMail IDs:

```
root@kuda[] #omadmidp -u -v
[OM 28612] 4 entries in the range requested are currently in use.
```

System Id	OpenMail Id.
10000	102
10001	103
10002	104
10003	105

```
Start of block: 10000
Range of block: 10
Used in block: 4
Start of block: 20000
Range of block: 10
Used in block: 0
Start of block: 40000
Range of block: 10
Used in block: 0
```

Warning! It is your responsibility to ensure that the pool IDs are not used by other Unix users. OpenMail will happily assign a pool Unix ID to a new OpenMail user without checking if it is being used by another non-OpenMail user.

If you try to add a pool ID user without having created the pool using `admidp`, or all the IDs in the pool are used, you get the following message:

```
root@kuda[sys] #omaddu -n "Mr Piglet/canberra,class" -a piglet -p piglet
omaddu : omaddu: [OM 8154] No more id's are available from the system id
pool.
```

Just for information, the file holding the ID pool is `/var/opt/openmail/sys/idpool`. It is a binary file containing pairs of 32 bit numbers. To see the contents of the pool file for the IDs above, I used the octal dump utility (`od`), requesting the output in decimal. I have underlined the entries showing the used IDs:

```
root@kuda[sys] #od -t d idpool
0000000      12345678          20          1          0
0000020          0          0          0          0
*
0000160          0          0          10000          102
0000200          10001          103          10002          104
0000220          10003          105          10004          0
0000240          10005          0          10006          0
0000260          10007          0          10008          0
0000300          10009          0          20000          0
0000320          20001          0          20002          0
0000340          20003          0          20004          0
0000360          20005          0          20006          0
0000400          20007          0          20008          0
0000420          20009          0          40000          0
0000440          40001          0          40002          0
0000460          40003          0          40004          0
0000500          40005          0          40006          0
0000520          40007          0          40008          0
0000540          40009          0
0000550
```

Changes To omaddu

```
Usage:  omaddu -n name/mailnode [-u UnixLogin] [-a alias]...
        [-c admin] [-l language] [-p password] [-e] [-R filename]
        [-v name]... [-V {+|-}secure] [-F]
```

omaddu has changed so that a Unix login name (-u) is no longer mandatory for the user being created. If this field is not specified, OpenMail will automatically allocate the first available Unix ID in the pool.

Remember that you will still need to include -u <Unix login name> when creating administrator users, as they need to be linked to a real Unix account.

(The -F option is to do with Outlook MAPI users. By specifying -F, an entry for the user is added to the FREEBUSY directory, so that they can share their calendar information. For more information on this, see the *OpenMail MAPI Service Provider Technical Guide* and the *Outlook MAPI OTN*).

Changes To omshowu

```
Usage : omshowu -m mailnode | all [-a] [-l]
        omshowu -n name[/mailnode] [ -G ]
        omshowu -U UnixLogin [ -G ]
        omshowu -O OmId [ -G ]
```

When you use omshowu to display user details, the Unix ID number chosen will be displayed. If the user has a password file entry, the Unix login name will be shown, otherwise the Unix ID number is displayed. Basically, omshowu gets OpenMail details and Unix ID numbers from the OpenMail USERLIST and Unix login names from the Unix password file entry.

In the following example, the user, normal, is an ordinary user, created with a pool ID:

```
root@kuda[sys] #omshowu -n normal -G
User Name : Mr Normal

MailNode : canberra,class

System Login : 10001

Password : set

Admin Capabilities : NO

Language : C

Internal user Id : 103

Aliases :
    normal

Access : Normal

Mail Account: Unlocked

Last Signon : 06.30.99 11:19:12
```

omshowu looks up normal in the USERLIST and finds his Unix ID number there, which it displays as System Login:. His OpenMail ID is shown as Internal user Id:.

Let's create an OpenMail user, Mr Rabbit, with administrator capabilities and assign him to an existing Unix login name, admin. The Unix ID number of the user, admin, is 5000:

```
omaddu -n "Mr Rabbit/canberra,class" -u admin -c admin -p 100acre
```

Remember to use the `-u` parameter. If you omit it, `omaddu` will assign a pool ID to the user and your admin user will not be correctly setup.

If I only have the Unix login name or Unix ID number of a user, I can use the `-U` option of `omshowu` to find out the details of the corresponding OpenMail user:

```
root@kuda[sys] #omshowu -U admin -G
```

```
User Name : Mr Rabbit
```

```
MailNode : canberra,class
```

```
System Login : admin
```

```
Password : set
```

```
Admin Capabilities : YES
```

```
Language : C
```

```
Internal user Id : 112
```

```
Access : Normal
```

```
Mail Account: Unlocked
```

```
Last Signon : 01.01.70 00:00:00
```

`omshowu` looks up the Unix login name in the Unix password file entry and gets the associated Unix ID number from there. It then uses the Unix ID number to look up the user's OpenMail details in the `USERLIST`.

```
root@kuda[sys] #omshowu -U 5000 -G
```

```
User Name : Mr Rabbit
```

```
MailNode : canberra,class
```

```
System Login : admin
```

```
Password : set
```

```
Admin Capabilities : YES
```

```
Language : C
```

```
Internal user Id : 112
```

```
...
```

In this example, `omshowu` first looks up the Unix ID number in the `USERLIST`, where it gets all the OpenMail details. It then looks for the appropriate Unix password file entry to find the associated Unix login name.

If I have the OpenMail ID of a user and I want to find out the Unix details associated with that user, I can use the `-O` option and give the OpenMail ID:

```
root@kuda[sys] #omshowu -O 112 -G
User Name : Mr Rabbit

MailNode : canberra,class

System Login : admin

Password : set

Admin Capabilities : YES

Language : C

Internal user Id : 112

Access : Normal

Mail Account: Unlocked

Last Signon : 01.01.70 00:00:00
```

Again, `omshowu` goes to the `USERLIST` first to get OpenMail details, then it searches for a Unix password file entry to find the Unix login name. If it doesn't find one, it will assume this user was set up with a Unix ID from the pool (and therefore does not have a Unix login name). `System Login` : would then show the pool ID number and not a Unix login name.

Changes To Areas Which Required UNIX Login Names

The following areas in OpenMail used Unix login names and/or read the Unix user's password file entry:

1. The name given to user UAL trace files was derived from the user's Unix login name, for example, `~openmail/tmp/fredU.log`.
2. The name of the file holding user specific tweaks was derived from the user's Unix login name, for example, `~openmail/sys/user.cfg/fred`.
3. A Unix home directory was required to run `omadmin`.
4. OpenMail passwords linked to Unix passwords.
5. Logging Unix login names in the audit log.
6. `omaddu`, `ommodu`, `omshowu` and, therefore, the administrator user interfaces: `omadmin`, PC admin GUI and admin centre integration (ITA/ITO).
7. `Advmail/TI`
8. Print Server

Trace And User Tweak Filenames

To remove the need to lookup Unix login names for trace filenames and user specific tweak filenames, the filenames are now derived from the OpenMail ID number (Internal user ID output by `omshowu`).

UAL trace files will become `OM<om-ID><X>.<Y>`, for example, `OM112U.log`, for a full command/reply trace log file and `OM112C.trc`, for a symbolic trace file. In some situations, the log filenames will take the form, `UXU.<Y>` or `OM<Unix-ID><X>.<Y>`, where `<Unix-ID>` is the Unix ID number of the user. This would be the case if the client uses `ual.local`, or if a very detailed logging level is requested, as logging starts before the OpenMail ID has been found.

User specific tweak files are now named `~openmail/sys/user.cfg/<om-ID>`. When you install B.06.00, a script is run to convert all existing user tweak files from their old names to their new ones. During this process, the `-U` option of `omshowu` is used to map the old Unix IDs to OpenMail IDs.

Home Directory For `omadmin`

A home directory will still be required to be able to run `omadmin`. OpenMail admin users will therefore need to have a Unix account configured correctly on the system.

OpenMail Passwords Linked to Unix Passwords

The `general.cfg` tweak, `UAL_UNIX_PASSWORD`, controls the link between users' Unix passwords and their OpenMail passwords. The default setting of this tweak is:

```
UAL_UNIX_PASSWORD=FALSE
```

which means OpenMail passwords are not linked to Unix passwords. Password aging for OpenMail passwords is set using entries in the `general.cfg` file.

If this is set to `TRUE` on your system, then your OpenMail users will need to be set up as proper Unix users with a Unix password file entry. Password aging, in this case, is controlled from Unix, not OpenMail.

Auditing

At B.05.10, setting auditing for user signon/signoff meant that the user's Unix login name was logged. At B.06.00, this has been changed to log the user's Unix ID number, as shown (underlined) in the example below:

```
root@kuda[] #omconfaud rci 1
omconfaud : Logging level updated OK.
root@kuda[] #tail -f ~openmail/logs/audit
user-signon
time 933240507 Thu Jul 29 10:28:27 1999 +60
user 102 Mr Admin/canberra,class 5000 5000
signon-status 0
```

To avoid a problem with backwards compatibility, a new tweak:

```
AUD_LOG_UX_NAME=FALSE
```

is available. This causes the Unix login name to be logged.

Administration UIs

As already mentioned, the Unix user ID (and Outlook changes) in B.06.00 have added several options to the `omaddu` (and `ommodu`) commands. These are now supported by `omadadmin` and the PC Admin GUI as follows:

omadadmin - Adding a User

If no Unix login name is entered when adding a user, `omadadmin` will attempt to use a pool ID. If the pool does not exist or is full, then an error is returned. If a Unix login name is entered, it is used, as before.

Once the user is added, an entry is added to the FREEBUSY directory, that is, the `-F` option of `omaddu` is assumed.

omadadmin - Modifying A User

If the user has a pool Unix ID number, their Unix ID is displayed as a number. If you modify the entry and enter a Unix login name to link the user to a real Unix account, the pool Unix ID is deallocated.

PC Admin GUI - Adding a user

If you choose the option to create a new Unix ID when adding a user, the system will attempt to use a pool ID. If the pool does not exist or is full, it will automatically revert to using the old method of adding a real Unix ID.

As with `omadadmin`, once the user is added, an entry is automatically added to the FREEBUSY directory.

PC Admin GUI - Modifying a user

If the user has a pool Unix ID number, the Unix Login ID will be displayed as `SYSPOOL`. If you change the Unix ID to a real Unix login name, the pool ID will be deallocated.

ITO/ITA

ITO/ITA have not been updated to take advantage of the the Unix ID changes. Therefore these always require a Unix login name to be assigned to each user.

Advmail Terminal Interface

To be able to use Advmail/TI, users must be able to logon and must, therefore, have a Unix login name. If you don't specify an OpenMail user name when you start Advmail/TI, it looks up the current user's Unix ID to find the corresponding OpenMail user. For this to work, you have to have created an OpenMail user with the associated Unix user's login name assigned.

If you are an administrator running Advmail/TI, you can use it to access the mailboxes of other users configured with pool IDs. For example, if Mr Normal is an OpenMail user with a pool ID, you could invoke Advmail/TI as follows:

```
advmail "Mr Normal"
```

Print Server

When the print server is run it runs a process under the pool ID assigned to the user. This works on most systems but some printing systems might not like the fact that no Unix user ID can be found.